CONTENTS

17. Dr. Thomas Marill (NIC 5656)
    Computer Corporation of America

18. Professor John McCarthy (NIC 5657)
    Stanford University

19. Mr. David McIntyre (NIC 5658)
    University of Illinois

20. Professor Marvin Minsky (NIC 5660)
    Massachusetts Institute of Technology

21. Professor Alan Perlis (NIC 5661)
    Carnegie-Mellon University

22. Dr. Bertram Raphael (NIC 5662)
    Stanford Research Institute

23. Dr. Robert Rapp (NIC 5647)
    The Rand Corporation

24. Professor Daniel L. Slotnick (NIC 5663)
    University of Illinois

25. Mr. Keith Uncapher (NIC 5664)
    The Rand Corporation

26. Mr. Clark Weissman (NIC 5665)
    System Development Corporation

1970 Project Activity: Augmentation Research Center (ARC)

D. C. Engelbart Stanford Research Institute

New Computer. Our XDS 940 has been replaced by a PDP-10 with a core memory that has twice the speed and three times the capacity (128K x 36 bits x 1 μs.), plus time-sharing hardware and software (TENEX) considerably more efficient than the 940. The PDP-10 will be a significantly more powerful support and will be readily expandable if more capacity is needed (the 940 was not expandable).

Improved System Languages and Architecture. The conversion of NLS to the PDP-10 occasioned an up-grading of both the programming languages we developed and of the software architecture for NLS. As a tool for systems programming, L10 (as we call this next-generation language) is significantly more powerful than the special-compiler language (MOL) we developed for the 940. It is also much less machine dependent -- the next time our large, complex systems are transferred to a new machine, we will be able to do all the reprogramming with a few man-months effort. The organization of NLS has been rearranged to derive a number of benefits. A very important one sets us up so that we can gracefully develop service for a wide variety of remote CRT terminals, through the ARPA Network, and with the full power of NLS. The combination of improved language and architecture will allow easy and rapid (explosive) extension of user features.

Connection to the ARPA Network. We have had our hardware connection since last summer. In our heaviest mode of use, we developed programs in Utah's PDP-10, and, as part of the conversion process to our new computer, we debugged PDP-10 versions of our NLS programs by shipping binary files to Utah's 10 via the Network (a matter of seconds). Our programmers operated Utah's debugging system from our own display consoles, with our 940 linking the console through the Network to the other computer, so that the programmer interacted just as though he were at Utah. (Our next TENEX NCP should be installed and operating by the end of February.)

Network Information Center (NIC). We have begun operating by telephone and the U.S. Mail. We established a "network" of R & C (Reference and Communication) Stations, one per Network site, holding hard copy reference material that we supply (by mail), and served locally by a Reference and Communication Agent assigned by the site. We set up "Enterprise" telephone circuits, covering the geographical areas of all sites, connected to two incoming private lines at ARC that are attended by an answering service. The system provides toll-free around-the-clock communication from all sites. A selected sub-collection of our master collection, including some 120 of our most relevant documents, has been replicated and a set installed at each Reference and Communication Station, together with a computer-generated hard-copy shelf list and author-sorted index. We support the interchange of memos and messages. We catalog them for future retrieval and, for the time being, distribute copies of each site's collection to stimulate dialogue.

On-line services will be available beginning this spring, including automatic message sending and cataloging, interactive querying, full-text retrieval of computer-held information, private-collection management, documentation-development aids, publication support, etc.

Remote Display Terminals. To expedite remote Network participants' use of NLS, we acquired an IMLAC self-contained processor-display system. For a little less than $20K total cost such a unit can be equipped with 8K of core, and outfitted with ARC's standard mouse and one-hand keyset to supplement its keyboard. We have installed a unit at a site 100 miles from our laboratory and connected it over a 2000-baud phone line. A top systems programmer works (for us) full time with the modified IMLAC -- currently using it only as a very fast typewriter, but steadily deriving advantage from its local storage and processor. We expect to achieve full NLS function within several months. Any Network site installing a similar IMLAC terminal could have full NLS (with fast response) very simply.

Dialogue Support System (DSS). This system of tools, files, conventions, etc., provides a means for accumulating, retrieving, and studying the collaborative communiques generated within ARC. Internal messages and memos, outgoing correspondence, official plans and designs, policies, reference guides, etc., as produced by our computer aids, are stored as computer files in retrievable form. Each entry is given a serial accession number and is cataloged as an entry in our one master catalog. Extensive techniques are being developed to facilitate search and retrieval (e.g., interactive set manipulation). The DSS has been developing steadily over the past eight months and is intended to be the means for everyday collaborative dialogue among ARC's system-development team. From DSS will come a facility for "backward" detection and pursuit of reference links. In scanning a catalog entry (for a file, a book, etc.), or an on-line file, the computer will be able to detect the presence elsewhere in the files of a link (reference citation) to that item, and it will know who generated the link and when. This back-linking facility will greatly enhance the power of our computer aids in studying and analyzing our on-line documentation and bibliographic files. (NOTE: When these DSS techniques become well checked out within our own working environment, they will be made available to support dialogue among other Network participants.)

RESEARCH IN STORE AND FORWARD COMPUTER NETWORKS

NETWORK ANALYSIS CORPORATION

SUMMARY

Technical Problem

The ARPA Computer Network will provide communication paths between computers distributed across the United States. The ARPA Contract with the Network Analysis Corporation involves the analysis and design of this network and the study of properties of networks of this type.

During the reporting period, the technical problems considered were the relationship between traffic, routing, throughput and cost. The main problems were to determine the effect of routing on network throughput and to study the properties of large store-and-forward networks.

General Methodology

The general approach to the above problems was to develop a powerful set of computer programs to optimize computer network cost and performance. The programs were used to derive estimates of optimum performance for specified networks, and to derive the tradeoffs between cost and throughput as a function of the number of nodes in the network under consideration.

Technical Results

A number of important technical results were derived during the reporting period.

1. A computer program that can find optimal routes for all traffic was developed. The program was used to derive upper bounds on performance and it was shown that the methods used to design the ARPA Network yield near optimal results.

2. It was shown that the performance of the ARPA Network is highly insensitive to input traffic. Therefore, if traffic forecasts are inaccurate, the network will retain its high throughput capabilities.

3. Cost-throughput relationships were derived for networks with 20, 40, 60, 80, and 100 nodes. It was shown that large networks are economical to operate using the present technology of the ARPA Network.

Department of Defense Implications

The substantial cost advantages of computer-communication networks with as many as 100 nodes was demonstrated. This property is of particular importance in satisfying the Defense Department's computer and communication requirements. The technical results imply that large computer networks can supply rapid and economical means for resource sharing and communications.

Implications for Further Research

The use of equipment presently being developed and communication line options not yet available may further enhance

the economy of large store-and-forward networks. This area should be pursued since by using the network optimization approach, it may be possible to suggest the most efficient equipment to develop.

UCSB RESEARCH REPORT SUMMARY

A. Network Development

The UCSB system development has been concentrated in two major areas: On-Line System software revision and ARPA Network development. With respect to the On-Line System, all goals outlined in our technical reports were met or exceeded. The new version of our software was released 1 July 70 as scheduled. Reliability immediately improved. Currently software failures are virtually non-existent. The system is completely exportable, and several computer centers have indicated an interest in obtaining our system for their installation.

Network development has proceeded at a rapid pace. A Network Control Program (NCP) has been written which supports the Host-to-Host protocol of 3 August 70, and makes the full capabilities of the network available to any assembly-language program which executes in the 360/75. Assembly-language subroutines have been written for call by Fortran and PL/1 programs, extending network accessibility to other than systems programmers. Network operators have been written for and added to the developmental version of the UCSB on-line system, enabling users to invoke the services of the NCP from an on-line environment. A network interface to the UCSB on-line system has been written, and accepts keyboard-like input over one network connection and returns alphameric, curvilinear, and special character output over another. This interface is invoked through the Logger, and specifications have already been distributed to members of the Network Working Group. Work is now in progress to develop a remote job entry facility which other nodes in the network may use. The card image input has been completed. Data set writers are not being written to process RJE output.

With the exception of jobs run via the network by RAND, all of UCSB's network experience has so far involved connection with both ends in local processes. UCSB is anxious to conduct tests involving NCP's at other sites, and will upon request, write short test processes to suit specific sites - establish connections with specified sockets, perform specified data transformations, etc.

B. Speech Research

During the past year the speech research effort has: 1) developed transformations which relate the ASCØF representation to the classical formant model of speech. These transformations have yielded good analytical estimates of formant frequencies and bandwidths. Techniques have also been developed for extracting the smoothed spectrum directly from ASCØF parameters, as an alternate to processing the speech by FFT cepstral methods. 2) defined a proprocessing method to filter the raw-speech string into sub-strings amenable to wave-function analysis. This has been accomplished using high-speed digital octave filtering algorithms. On the IBM 1800 the total preprocessing time, including extrema determination, is now 35 sec./sec. of speech and could be reduced to 16 seconds with installation of a good analog low pass filter on the input. 3) developed a one-pass wave-function analyzer based on the Gaussian Cosine Modulation Model. The current analyzer provides excellent speech fidelity and takes 2.8 sec./sec. of speech. 4) classified and recognized phonetic information including extraction of recognition

parameters from the ASCØF parameter set. This research has established that recognition parameters can be extracted from the ASCØF data and reliable recognition performed on a subset of phonemes. A segmentation algorithm has been developed which reliably partitions speech into voiced-unvoiced segments, and identifies transitional voiced regions as distinct from steady-state vowel-like regions. Using this information, a single speaker vowel recognition algorithm has been implemented to identify the ten vowels and two dipthongs, when spoken between two unvoiced consonants. Although many vowel recognizers have been developed, it is significant that the ASCØF representation provided a highly satisfactory basis for this current effort. Moreover, the segmentation and recognition techniques employed form an essential first step toward a general recognizer package based on ASCØF data. 5) studied the data rate of the basic ASCØF representation and the amount of data compression possible by elimination of redundant parameter sets and encoding techniques. Preliminary studies have shown that the basic bit rate of 40-50,000 bps can be reduced to approximately 10,000 bps with little or no loss in perceptual quality, and have indicated that further gains are possible through predictive encoding techniques.

SUMMARY REPORT
University of California at Los Angeles
by
Leonard Kleinrock, Principal Investigator
Contract No. DAHC-15-69-C-0285

The principal activities associated with the ARPA contract at UCLA are: network software; network measurements; and computer systems modeling and analysis (especially computer networks and time-shared computer systems). Our principal efforts in these areas for this past year are summarized below.

During this year, we took on the major task of establishing the HOST-HOST network protocol. This effort was a coordinated negotiation among many sites finally producing a well-specified network protocol; this protocol has since been implemented at many sites into an appropriate NCP. Our own NCP has just come up and is appropriately imbedded in our system executive (SEX) time-sharing system (128K virtual user memory in a 32K physical core!).

As Network Measurement Center, we have created software which formats the measurement data directed to our node at UCLA by any IMP which is so instructed. This data is composed of histogram data on average statistics, snap-shop data on instantaneous conditions within an IMP, and trace data which tracks the history of a particular message as it passes through the net. Numerous network measurements and experiments were carried out to investigate the network performance and also to test the measurement tools themselves. These tests were primarily concerned with message delays, throughput, routing, and, in the case of the SRI-Utah traffic measurements, with the distribution of message parameters for real HOST-HOST traffic. Among the observations we made, we were able to establish that during a Utah to SRI file transmission, the effective data rate was within 10% of the theoretical maximum data rate (approximately 30 kilobits of data per second).

By far, our most intense effort took place in the area of modeling and analysis of computer systems. In the field of computer networks, our modeling effort has gone rather far. We now have an analytical model for the utilization of nodal storage within ARPA-like computer networks. Comparison of analytical results with simulation results is quite good in this rather difficult area. Furthermore, we have characterized the segmenting of messages which takes place when they are broken into packets and are able to make theoretical predictions regarding the inter-packet delay during message re-assembly time. Also, we have made some careful studies as to how the message traffic should be broken into priority groups (where short messages are given higher priority than longer ones) and have located optimal thresholds for the priority determination. We have reported in the literature on a study of more realistic cost functions in the general computer network design problem and have shown for the ARPA network that power law cost functions (which fit the available tariff data reasonably well) are accurately modeled by a linear cost function as regards the average message delay under the optimum channel capacity assignments. The very nasty problem of blocking which takes place in networks (that is, when the storage within an IMP fills and "blocks" other messages from entering) and the propagation of this blocking effect to neighbors is currently under study. We have created a model and an analysis of that model which allows us to predict the fraction of blocked nodes within large networks as a function of time and also to estimate clump sizes for these blocked groups. We have been studying

multiplexing techniques for concentrating data from terminals to computers over high-speed channels (in particular, asynchronous time division multiplexing). We have also been looking into the general multi-commodity flow problem and have come up with algorithms which are rather efficient; we are now tooling up to handle non-linear constraints which typically arise in communication networks. In that same vein, we have devised iterative techniques for simultaneously adjusting the routing matrix as well as the channel capacity assignment to obtain good (not necessarily optimal) network designs; these efforts have been especially rewarding in their efficiency and simplicity.

Our second modeling area has also been extremely rewarding in the richness of its results; this is in the area of time-shared computer systems modeling and analysis. We have created an entirely new class of scheduling algorithms which permits an arbitrary mix of first-come-first-serve, round-robin, and foreground-background disciplines where a job is treated according to many of these disciplines depending upon how much service it has already received. We have also created a continuum of scheduling algorithms (along with the pertinent analysis) whereby we can vary simple parameters in this scheduler and effect a change from the non-discriminatory first-come-first-serve algorithm through to the moderately discriminatory round-robin system all the way to the completely discriminatory foreground-background system (the discrimination is, of course, based on job length). Thus we have smooth functions which permit us to range over an entire domain of interesting scheduling algorithms. In fact, there are so many available and analyzed scheduling algorithms, that we have faced the question of bringing some order into this embarrassment of riches. In this endeavor, we have been particularly successful and have established certain necessary conditions on response time functions (e.g., a monotonicity property and also a conservation law on the work in the system). More importantly, we have been able to derive tight bounds on the response time functions for any scheduling algorithm which does not make use of a-priori knowledge of service times; we are able to synthesize schedulers such that they touch any point on these upper and lower tight bounds for performance.

In addition to the three major activities reported above (network protocol, network measurement, computer systems modeling and analysis), we are engaged in a variety of other minor but interesting activities. Among these is the software development of a new display system which we call "fisheye." In this system, large graphical data structures may be displayed in a non-linear fashion such that in the central region, magnification and linearity of display are obtained and, as one moves away from the center, compression (and attendant distortion) permits one to squeeze the entire display set into the physical confines of the display system. The advantage of this form of display, of course, is that one may see great detail in an area of interest, and move that area while maintaining peripheral vision with regard to the global data set.

THE RAND CORPORATION
1700 Main Street
Santa Monica, California 90406
(213) 393-0411

OVERVIEW OF RAND'S IPT RESEARCH IN MAN-MACHINE SYSTEMS

The aim of this research is to provide the basic computer science necessary to produce user-oriented systems that are directly applicable and beneficial to the military. The following described programs involve the study, definition, and prototype implementation of hardware systems and software system structures which, together, are flexibly and easily adapted to specific end user needs.

ARPA EXPERIMENTAL COMPUTER NETWORK (J.F. Heafner, E.F. Harslem)

RAND has developed programs to enable its researchers to use remote resources, and has worked with the Network Community to develop common software.

Host software has been verified which will extend message passing to include paths between the IMP and service machine, containing the Network Control Program (NCP). We have verified our NCP with UCSB and will assist other sites in their NCP checkout.

Experimentation and services have been combined into a common program, the Network Services Program (NSP). Presently we allow any Video Graphics System (VGS) console user dynamic access to data sets and to remote programs.

We plan to continue NSP development to include program development remotely, automatic data reconfiguration, remote job access, and graphical applications for RAND researchers.

We have provided specifications for a "form machine" addressing these problems.

AN ADAPTIVE COMMUNICATOR (R.H. Anderson, W.L. Sibley)

The Adaptive Communicator (A/C) Project is an attempt to make the interface between man and machine more flexible and man-oriented. It is based on three design features: (1) the communication task is separated from the application program; (2) multiple communication channels can be used simultaneously and interchangeably; (3) the communication agent is adaptive; its behavior is both "trainable" and easily modified.

The internal structure of the A/C consists of several cascaded levels of table-driven recognizers. Perhaps the most important aspect of this project is the development of a consistent notation for representing action-in-a-context during man-machine interaction; this notation consists entirely of sets of pattern-replacement rules with both the pattern and replacement parts consisting entirely of sets of symbols with attached attributes. This notation permits adaptive behavior through automatic generation of "learned" pattern-replacement rules and the subsequent "blending" of these rules into existing rule sets. Most aspects of the system now exist in prototype form. During the next several months we intend to investigate and test several different modes of adaptive behavior which are possible in the A/C notation and system structure.

COMPUTER-ASSISTED SPECIFICATIONS OF ALGORITHMIC PROCESSES (R.M. Balzer)

This is a long-term project aimed at understanding and defining the semantics of instructional conversation, and the mechanisms required to execute their instructions so that a computer can be instructed to perform desired tasks. The ultimate form of this specification will be as close as possible to the form used between two intelligent humans.

CASAP is based on the use of "context," assimilation of new concepts, dynamic process adaptation by modification, system maintenance of "housekeeping" data structures, and normal-case specification of processes.

The internal data representation for both program and data, consisting of directed graphs containing nodes and edges which are English words, has been implemented. The input, dictionary building, and information management routines have been completed and debugged. The graph matching, graph executor, and context extractor routines have been coded but must be debugged and integrated. Set representations and manipulations have been chosen as the basic semantic actions which the CASAP interpreter can perform and understand.

We hope to complete the initial version of CASAP during the next reporting period and concentrate on: integrating separate process specifications; dynamic modification of these specifications; and inference and maintenance by CASAP of housekeeping data. Our views on program organization, upon which CASAP is founded, will be published as R-622-ARPA.

STUDIES IN COMPUTER PROGRAM ORGANIZATION (J.C. Shaw)

The goal of this project is to endow the user's agent (i.e., the interpreter of problem-oriented languages) with judgment and problem-solving power. The user would then be spared the programming of complete procedural specifications to the extent of the agent's ability to discover procedures to accomplish higher level tasks.

We have worked toward the solution of several central problems:

1. How do we get the agent to judge alternative actions and choose one?

2. How do we organize the agent to automatically forget less essential information in order to stay within the limits of high-speed storage?

3. What data structure and processes are appropriate to represent the agent's imagination as it attempts to foresee the consequences of alternative actions without actually performing the action in the real situation or in an external model of it?

4. How do we organize the agent to give deep consideration to difficult decisions of great consequence and range down to almost immediate decisions in simple situations?

We intend to sharpen the notion of judgment in a computer in order to make feasible the programming of an agent to demonstrate this concept.

COMPUTER APPLICATIONS TO MILITARY PROBLEMS (K.W. Uncapher)

The three primary goals of this project are to: (1) inform the military of current computer work that has potentially high payoff for direct military applications; (2) assist the military in effective transfer of technology and methodology from the research community to the military; and (3) shape research projects toward military needs.

We are assisting the Defense Intelligence Agency and the U.S. Air Force Rome Air Development Center in the design of a video hardware-software system for DIA. Currently, we are providing similar assistance to Air Force Global Weather Central. We plan to assist GWC in the design and installation for an on-line graphics system to be used in the generation of weather maps and in the distribution of weather map annotation to base-level weather status once the user needs are completely defined. Through these efforts a mechanism now exists whereby products from the ARPA research efforts can be made available to the operationally-oriented military users.

The Defense Science Board's report on computer security, edited by W. Ware, is now completed and is being distributed. One hundred copies have been delivered to the DSB for its distribution, and 500 copies of the RAND edition have been distributed to interested addressees in military organizations and in industry.

The remainder of the work in this area is classified and, therefore, cannot be reported herein.

IPT EXPLORATION RESEARCH AND SUPPORTING EFFORTS (T.O. Ellis)

This program investigates new areas of research appropriate to the total IPT program at RAND, and provides specialized hardware and software systems to enhance the on-going projects.

The principal efforts in progress are:

1. An increase of the interactive terminal facilities available to IPT research projects. Twenty-five full graphics consoles are now operating in the VGS environment.

2. Definition and design of a computer system to support programming research. The project's approach is with a dual processor computer system wherein the access, protection, and I/O facilities are provided by adaptation of existing stable software and hardware as a controlling engine for a second, very flexible, microprogram processor. The second processor's definition can be switched at high speed for task-oriented addressing structure, word structure, instruction codes, and operand conventions.

3. Implementation of voice-directed input. The intent of this effort is to obtain a practical system for limited vocabulary (50 to 200 phrases) to study the value and needs of such communication, using operational on-going systems as a vehicle. Initial trials on a VGS console have provided encouraging results.

SP-3585

SP *a professional paper*

This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract No. DAHC15-67-C-0149, ARPA Order No. 1327, Amendment No. 3, Program Code No.1D30, and 1P10.

SDC 1970 PROGRAM OVERVIEW

COMPUTER ASSISTED PLANNING (CAP)

C. Weissman

SYSTEM
DEVELOPMENT
CORPORATION
2500 COLORADO AVE.
SANTA MONICA
CALIFORNIA
90406

SDC 1970 PROGRAM OVERVIEW

COMPUTER ASSISTED PLANNING (CAP)

INTRODUCTION

Most of the active projects reported here have the long-range goal of substantially improving understanding of the strategic planning process and of embodying that understanding in an experimental prototype computer-based system. The system envisioned would integrate old and new technologies and would permit teams of DoD planners to interact, via natural communications, with each other, with computer-based data bases, and with analysis tools to achieve planning objectives more rapidly or with higher quality.

For the short range, SDC will examine the computer and communications foundations and trade-offs for such a planning system. Communications is taken in its broadest sense--man/machine (programming, command, and English languages; graphics; and voice input/output) and machine/machine (networks, compiler optimization, computer-communications trade-offs, and time-sharing). Studies of problem solving and learning by man-machine teams are probing for an improved organizational basis for computer-assisted planning--a nominally unstructured problem--by use of interactive problem-solving techniques in which the computer is an interactive and intelligent partner in the synergistic solution process.

For a more complete project description and bibliography, see TM-3628/007/00, Computer Aided Command, September 1970. Many of these projects are currently inactive due to funding constraints.

CONVERSE

This project, directed by Charles Kellogg, is concerned with constructing and exercising a vehicle for user communication in ordinary English with large, on-line data bases. Specific aims include: (1) implementation of a comprehensive lexical recognizer; (2) implementation of a comprehensive parsing technique; (3) implementation of a natural-language-oriented data-management system; (4) development of a viable approach to carrying out useful deductions over large data collections; (5) implementation of techniques to ensure user confidence in system output; (6) implementation of techniques to facilitate user English subset extension; (7) active collaboration with other allied research efforts.

Progress in 1970 may be summarized as follows: (1) A lexical recognition routine that uses an efficient hash coding technique to associate lexical items with concepts of vocabularies as large as 25,000 items was checked out. (2) Major improvements in the parser include the development of a new approach to simultaneous production of all surface and deep structure parsings and semantic interpretations by use of a set of general structure-building and structure-changing rules. (Kellogg, et al, SP-6534, in preparation.) (3) A data-management system has been

implemented and is in use with two data bases of 4,000 and 10,000 facts (N-tuples) respectively. (4) A promising new approach to large-scale data base inference making has been developed. (5) Logic for extensive user guidance has been worked out. (6) User-extendibility techniques currently include an initial limited capability to introduce new terms, concepts, and modes of expression. Coding of a more advanced capability will start shortly. (7) Collaboration with John Olney, UCLA, has produced a complete morphographemic description of the productive suffixes of English, initial semantic data on case relations, and a reformulation of Olney's rules for resolving anaphora. Cooperative activities with the SDC Voice I/O project is increasing.

GRAPHIC I/O.

This project, led by Mort Bernstein, is concerned with improving man-machine communications through the development and use of highly interactive computer graphic input/output technology. Through the use of an interactive display and associated data tablet input device, the goal is to develop the tools and systems necessary to allow use of the common 'pencil and paper' notations for computer programming and scienticic problem solving.

In past years, the following systems were developed on the IBM AN/FSQ-32: a character recognizer, providing recognition of a large alphabet; a parser which converted ordinary two-dimensional mathematical notation to a linear string form; and an unparser which converted the linear form back to the textbook quality mathematics. In 1970, most of these programs were converted to the IBM 360 ADEPT time-sharing system, and a new improved display was fabricated and placed into service. In addition, the following tasks were performed: (1) Implementation of a new 360 assembler. This assembler (Howell, TM-4550) includes a version of the L^6 programming language and a partial reassembly feature allowing a single routine in a program to be replaced by a new or modified version of that routine without reassembling the entire program. (2) Initial design, coding, and test of significantly improved character recognizer and dictionary builder for both the IBM 360 and the DDP-516 programmable controller. (3) Initial design and coding for a numeric interpreter for the output of the parser. This is the first step toward a full computational facility using mathematical notation. This interpreter includes function definition and expressions providing most of the usual computational and matrix operations. Specifications for a Flowchart Programming System (Bernstein, TM-4582) have been published and work has begun on the basic routines.

GRAPHIC TABLET DISPLAY CONSOLE DEVELOPMENT

In conjunction with the Graphic I/O project, a prototype graphic tablet display console was developed in 1967 which demonstrated the feasibility of creating a single interactive surface by projecting an interactive CRT image onto the rear of the RAND tablet. This console permits user interaction with the computer in a very natural 'pencil and paper' mode. Present work, led by Tom Williams, is primarily concerned with refining and improving the concept using a Beta Instruments Corporation display with very fast short-vector capability. The display and tablet

are interfaced to the ADEPT Time-Sharing System through the DDP-516 programmable controller, permitting user control over the interface software. Possible future activity is development of a minicomputer-controlled multi-terminal system with hand-printed input and graphic output.

DISPLAY-70

Display-70, under the guidance of Tom Williams, was completed successfully during the year. It explored the feasibility of building an interactive data base retrieval and data presentation system using a storage tube graphic display. The resultant system provides the ability to select data and display it in a wide variety of tabular and graphic forms and to change or save these forms with little or no penalty. The system achieves independence of a particular data base system or structure through the use of an "Ancestor file" from which all work proceeds. After an Ancestor file is constructed, the user may select from a variety of display options, specifying subsetting operations on the data prior to display. The user is not constrained by prespecified operation sequences and is free at any time to abort or save a presentation for future use. Basic statistical functions may be computed from these data and the user may choose to have a first-, second-, or third-order polynomial fit to his data and display. Although development of this system ceased in September, it is being used experimentally by the CACTOS project.

VOICE INPUT/OUTPUT

The Voice I/O project, headed by Jeff Barnett, is investigating and developing systems for man-machine interaction using continuous speech. The initial target product is a limited vocal data management system having syntactic constraints, good semantic prediction capabilities, and a vocabulary whose size and complexity may be varied.

The Stanford speech understanding system, by Vicens and Reddy, is being reimplemented and extended at SDC to serve as a first-level routine for the acoustic recognition tasks (Kameny and Ritea, TM series 4652). Hardware fricative detectors and classifiers and other software feature extractors will be added to the system. Phonetic classification rules will be implemented in a specially developed symbolic language for flexibility in our experimentation. In parallel with these improvements, techniques for interfacing and higher-level component specifications are being investigated. Information paths and program-control flow monitoring represent the dominant problems. A set of small system performance test vocabularies containing a particular kind of recognition conflict is being designed. As changes are made to the system these vocabularies will be used to determine where improvements and regressions occur. In addition, instrumentation will be included to determine where computation time is being spent and how best to proceed.

A sound laboratory, built around a Raytheon 704 computer and high-quality recording playback facilities, has been completed recently and experimentation has begun.

NETWORKS

The Network project team, led by Bob Long, merged the subsystem HOSTOSS into the ADEPT time-sharing system to connect it to the ARPA network. HOSTOSS is designed to permit remote network users to LOGIN and run existing SDC programs under ADEPT without the need to modify them. It also enables local users to establish connections to remote systems and it provides for local and remote interprocess communication. The network Interface Message Processor (IMP) is connected to the ADEPT system on the IBM 360/67 through the Honeywell DDP-516 programmable controller. Therefore the software interface includes a DDP-516 program called HOSTOSS/516 and a set of IBM 360 programs called HOSTOSS/360. HOSTOSS/516 interacts with the IMP and with HOSTOSS/360 via the ADEPT terminal I/O handler, which also has software components in both computers. HOSTOSS/360 interacts with ADEPT and performs the function of network-control program (NCP) in compliance with the HOST-to-HOST protocol. Production of the HOSTOSS/516 test programs, and a simplified version of the HOSTOSS/360 was completed with receipt of our IMP in April 1970. The entire HOSTOSS/360 system is in final checkout for February operation.

Arie Shoshani is investigating the problem of data management in computer networks. Various issues are being considered, including central vs. distributed data bases and homogeneous vs. heterogeneous data-management systems. Practical considerations include cooperative short-term usability and ease of implementation. Other network applications being examined include cooperative research in voice, English, and graphic research.

META-GRAPH ANALYSIS

This project, led by Erwin Book, is concerned with the application of graph theory to compiler optimization. Project goals include producing descriptions of optimization algorithms in mathematical terms, proving important theorems, defining a language for expressing the algorithms defined, and using this language to implement these algorithms so they run on a computer.

In conjunction with extensions made to the Meta-Graph compiler by Val Schorre to make it more powerful in the area of compiler optimization, work has begun on a FORTRAN IV compiler as a benchmark illustration of these optimizing techniques. Improvements in compiler-compiler technology have resulted from a study of time and space requirements for data structures used in compiling and of the feasibility of performing the desired optimization on a computer with medium-size memory. Marvin Schaefer has developed the theoretical foundation on which the FORTRAN compiler stands. The formal model represents computer programs by a series of finite directed graphs, the simplest of which is a control-flow graph similar to a flowchart. An original comprehensive text resulting from this activity on the state of the art of compiler optimization is nearly complete.

CACTOS (COMPUTATION AND COMMUNICATION TRADE-OFF STUDY)

This project, headed by Joe Robertson, was initiated in September of 1970 to investigate means of minimizing a possible mismatch between DoD computer and communication requirements and the technology of the 1975-80 era. Minimization is to be effected by trade-offs between computation and communication capabilities.

The CACTOS team is surveying the technology and developing forecasts. Simultaneously, they are surveying those critical DoD agencies and organizations that are repositories of summary information of the systems, communication, and computer plans and requirements of operational military and governmental agencies.

Using gross estimates of major technological trends and requirements, several analytic models were developed, programmed, and run during the first three months of the study. These models were used to perform a sensitivity analysis of important system variables, i.e., load, capacity, cost, the economies of scale, response time, and system configuration. An interesting method of using decision trees of "alternative futures" to evolve critical trade-off curves has been developed by the study team. These will be elaborated as the study progresses to determine cost-effective ratios of configurations and where emphasis should be exerted on the technology to achieve cost-effective system design. In addition, attempts are being made to characterize computer-communication networks formally in terms of their principal variables of interest, e.g., geometry, complexity, capacities, etc.

ADEPT EXECUTIVE SYSTEM EXTENSIONS

The ADEPT-50 executive (Linde and Weissman, 1969 FJCC Proceedings) functions as the operating system for all the SDC projects reported herein. This project, led by Jerry Cole, was concerned with research and development in executive systems, particularly to support networking and programmable communications controller experimentation. During the last quarter of the year, efforts focussed on mobility of ADEPT by transferring the system from an IBM 360 Model 50 to a 360/67.

The programmable controller development activity permitted investigation of a flexible hardware/software mechanism for interfacing new terminals, real-time display equipment, and the ARPA network. The hardware part of this sytem consists of a Honeywell DDP-516 connected to the multiplexor channel of the IBM 360 through a Honeywell-supplied interface. The transfer from the Model 50H to the Model 67I computer allowed improved services--swapping from the 2301 drum, larger programs (85 pages), and additional resident system functions for improved responsiveness.

SECURITY

This project developed practical security controls in multi-access systems (Weissman, SP-3342 and 1969 FJCC Proceedings). During the past year a comprehensive package of security-checking software was produced for the ADEPT TSS as an existence proof of novel security controls. SPY--the automatic surveillance function--conducts periodic assaults on ADEPT-50 and its environment in an attempt to compromise the integrity of the system. SOS--the Security Officer's Station--is

responsible for providing an interactive security surveillance capability within ADEPT-50. The security officer, operating from the SOS, is the agent who monitors, via the SCRIPT program, the on-going system activity and assumes complete responsibility for system security maintenance. This activity was successfully concluded in September with demonstration and documentation of these experimental mechanisms that contribute to system verification and certification.

SYSTEM TEST AND EVALUATION

The primary objectives of the System Test and Evaluation Project, led by Arnie Karush, was investigation of many techniques for system instrumentation, reliability monitoring and associated data reduction, and flexible and economic acquisition of data describing the behavior of computer systems. The activity probed five areas. (1) Regenerative recording was successfully investigated as a technique for real-time capture and non-real-time rerun of the complete behavior of ADEPT operation. (2) A theoretical model describing an instrumentation capability was developed and applied to the ADEPT-50 system. The capability allowed the user to instrument and de-instrument the system from an operational interactive terminal. (3) Instrumentation of user programs written in higher-level languages was explored, and measurement routines were produced for compile-time inclusion in JOVIAL programs. The resultant object program reported their self-measurement on the user's interactive terminal. (4) Two hardware monitors were attached to the ADEPT-50 system, enlarging our understanding of the strengths and limitations of this technique. (5) A previously developed time-sharing system bench-mark package was refined and improved. The DDP-516 programmable controller was programmed to behave as a "forcing function" to drive the ADEPT TSS with a simulated user load. This was achieved by running the bench marks automatically with scripted terminal inputs based upon user selected combinations of operating parameters. The project was concluded in September 1970.

FORMALIZED OPERATING SYSTEMS

The long range goal of the Formalized Ope-ating Systems Project, led by Alex Tschekaloff, was to lower costs for the design and fabrication of operating systems by using formal software engineering practices. Specific objectives included (1) the development of a formal description of operating system architecture, and (2) a functional and structural analysis of the logical parts of an operating system. Our approach was to design languages in which the system structure and function entities could be specified to develop a set of compilers capable of handling these languages, and to use these compilers in the production of an operational model.

This project is currently inactive, however progress was made toward our original goal by completion of the following tasks: (1) A first approximation to a language that describes the over-all structure of an operating system was designed. (2) A logical model of an operating system consisting of such parts as interrupt handlers, sequence control, input-output control, access methods, file maintenance, program loading, debugging, and program service routines was developed.

(3) Languages in which the sequence-control functions, the initialization function, and the system-control data can be expressed were designed. (4) Compilers capable of producing code for the sequence-control function were written. (5) A version of the compiler capable of syntax-scanning data structures for the data module language was written.

MAN--MACHINE SYNERGY

The long-range goal of this project, led by Aiko Hormann, is the development of a man-machine system that effectively couples the complementary capabilities of man and machine for cooperative planning and creative problem solving in practical real-world situations.

During the early part of 1970, the original design of Gaku, the machine portion of the man-machine system, underwent modifications incorporating desirable new features and remedying limiting factors uncovered during experimentation with previous designs. Emphasis was placed on providing assistance in man's planning and problem-solving functions and on developing a flexible man-machine communication. A detailed description of the design of a man-machine communication language can be found in "User Adaptive Language (UAL): A Step Toward Man-Machine Synergism," (Hormann, Crandell, and Leal, TM-4539). Implementation of the Gaku-UAL design on the IBM 360/67, using the SDC LISP system is in progress.